

A lineáris optimalizálás rugalmas indexválasztási szabályainak elméletéről és gyakorlatáról

Nagy Adrienn

A doktori disszertáció tézisei

Témavezető: Illés Tibor

Egyetemi Docens, PhD

Témavezető: Kovács Margit

A Matematika Tudomány Kandidátusa

Eötvös Loránd Tudományegyetem, Matematika intézet,
Doktori Iskola

A doktori iskola vezetője:

Laczkovich Miklós Egyetemi Tanár
a Magyar Tudományos Akadémia tagja

Alkalmazott matematikai doktori program

A doktori program vezetője:

Michaletzky György Egyetemi Tanár
a Tudományok Doktora

A tézis az Eötvös Loránd Tudományegyetem Operációkutatási Tanszékén
készült.

Budapest
2014. Május

A tézis célkitűzései

A tézis a lineáris megengedettségi feladat, a lineáris programozási feladat (LP), a lineáris komplementaritási feladat (LCP) illetve a lineáris feltételes konvex kvadratikus programozási feladatok (QP) pivot algoritmusainak, valamint rugalmas indexválasztási szabályainak elméletével és gyakorlatával foglalkozik.

A gyakorlati optimalizálási feladatok jelentős része erősen degenerált, sőt, a legtöbb valós feladat tartalmaz primál vagy duál oldali degeneráltságot valamilyen formában. A degeneráltság mindkét formája gyakori, de talán a duál oldali egy picivel jellemzőbb, hiszen tetőleges változó melynek nincs célfüggvény együtthatója a legtöbb pivot táblán degenerált lesz. A degeneráltság kedvezőtlen tulajdonság a pivot módszerek számára, hiszen degenerált táblákon az algoritmusok hajlamosak nulla hosszú lépéseket tenni mikor ugyanazon megoldás számos bázissal leírható. Azon szimplex variánsok esetén ahol a célfüggvény monoton, a gyakorlatban a végességet gyakran valamilyen perturbáció segítségével biztosítják. Az is elmondható, hogy a pivot eljárásokkal foglalkozó korai publikációk jellemzően perturbációs megfontolások segítségével indokolták az algoritmusok végességét. Egy indexválasztási szabály alkalmazása jelentős megkötést jelent a numerikus implementációk esetén, és így azok jellemzően egyéb hatékonysági és stabilitási megfontolásokat részesítenek előnyben. A numerikus hibák számos esetben természetes módon adódó perturbációs hatást okoznak, mely esetenként beavatkozás nélkül segít kilépni a degenerált megoldásból. Mindazonáltal, bármilyen jellegű perturbáció jelenléte vagy alkalmazása után, a kapott megoldás nem pontosan az eredeti feladat megoldása, így azt finomítani kell a perturbáció eltávolításával. Ez az eljárás egy úgynevezett másodlagos ciklizálási jelenséget indíthat el.

A rugalmas indexválasztási szabályok, mint amilyen például a leggyakrabban mozgó változó (MOSV) szabálya, illetve az utoljára mozgott változó (LIFO) szabálya ötvözik az elméleti végesség biztosítását a gyakorlati megfontolások szempontjából fontos rugalmassággal, melyek degenerált táblákon jellemzően nem határozzák meg egyértelműen a pivot pozíciót mikor annak választása az algoritmus által nem egyértelműen rögzített.

A tézis a numerikus vizsgálatok során a következő szabályokat vizsgálja a rugalmas indexválasztási szabályok gyakorlati megfontolásokkal való ötvözése során:

- *Hybrid-LIFO*: mikor a LIFO nem határozza meg a választást egyértelműen, a legnegatívabb szabályt alkalmazzuk,
- *Hybrid-MOSV*: mikor a MOSV nem határozza meg a választást egyértelműen, a legnegatívabb szabályt alkalmazzuk.

A rugalmas indexválasztási szabályok keretbefoglalására az [1, 2]-ben bevezetett **s**-monoton szabályokat alkalmazzuk:

1. Definíció (Lehetséges pivot sorozat [1]). *Indexpárok egy sorozatát*

$$\mathcal{S} = \{\mathcal{S}_k = (i_k, o_k) : i_k, o_k \in \mathbb{N} \text{ ahol } k \in \mathbb{N}\},$$

lehetséges pivot sorozatnak nevezzük, ha

- (i) $n = \max\{\max_{k \in \mathbb{N}} i_k, \max_{k \in \mathbb{N}} o_k\}$ véges,
- (ii) és létezik egy n változós lineáris optimalizálási feladat ahol $\text{rank}(A) = m$, továbbá
- (iii) egy (esetleg végtelen) pivot sorozat, ahol a mozgó változók indexei éppen az \mathcal{S} -beli pároknak felelnek meg.

A pivot sorozat a pivotálási sorrend és az alkalmazott pivot algoritmus indexválasztási szabálya között teremt kapcsolatot.

2. Definíció (Pivotálási index preferencia [1]). *Vektorok egy $\mathbf{s}_k \in \mathbb{N}^n$ sorozatát pivotálási index preferenciának nevezzük, ha egy tetszőleges j -edik iteráció során, ha a pivotálási szabály nem határozza meg egyértelműen a pivot pozíciót úgy az algoritmus azon indexek közül választ melyeknek a legnagyobb az értéke a lehetséges indexek közül az \mathbf{s}_j vektor szerint.*

Az \mathbf{s} -monoton indexválasztási szabály egy közös monoton tulajdonságot fogalmaz meg számos ismert szabály között.

3. Definíció (\mathbf{s} -monoton indexválasztási szabály [1]). *Legyen adott $n \in \mathbb{N}$. Egy indexválasztási szabályt \mathbf{s} -monotonnak nevezünk, ha*

1. *létezik olyan index választási preferencia és hozzá tartozó $\mathbf{s}_k \in \mathbb{N}^n$ melyre*
 - (a) *az \mathbf{s}_{j-1} vektorbeli értékek a j -edik iteráció után kizárólag az i_j és az o_j indexekre változhatnak, ahol az i_j és az o_j a pivot művelet indexei,*
 - (b) *az értékek nem csökkenhetnek, vagyis (nem feltétlen szigorúan) monoton növekedőek.*
2. *egy végtelen \mathcal{S} lehetséges pivot sorozat esetén, tetszőleges j -edik iteráció esetén létezik egy $r \geq j$ -ik iteráció, melyre teljesül hogy*
 - (a) *azon \mathcal{I}^* indexekre nézve melyek végtelen sokszor szerepelnek a \mathcal{S} indexek között, a $\mathcal{I}^* \cap \mathcal{I}_{B_r}$ indexhalmazon azon index melyre az \mathbf{s}_r érték minimális egyértelmű (legyen az az l index) ahol \mathcal{I}_{B_r} az r . iterációban a bázisban levő indexek halmaza,*
 - (b) *egy $t > r$ iteráció során ahol az $l \in \mathcal{I}^*$ az első alkalommal szerepel ismét az \mathcal{S} indexpárjai között, azon indexek melyek az \mathcal{I}^* halmazból előfordultak a \mathcal{S} indexpárjai között szigorúan a S_r és S_t -hez tartozó iterációk között, nagyobb értékkel szerepelnek az \mathbf{s} vektorban mint az l index.*

A tézis egyaránt tartalmaz elméleti és számítási eredményeket az **s**-monoton indexválasztási szabályokra nézve; azon algoritmusokra nézve melyekre a végesség már korábban bizonyításra került ezen szabályokkal a tézis elsősorban numerikus eredményeket mutat be, míg más pivot algoritmusokra először bizonyítja a végességet.

A criss-cross módszer

Legyen adottak a $A \in \mathbb{R}^{m \times n}$, $\mathbf{c} \in \mathbb{R}^n$, $\mathbf{b} \in \mathbb{R}^m$ mátrix és vektorok a megfelelő dimenzióval; ekkor

$$\begin{aligned} \min \quad & \mathbf{c}^T \mathbf{x} \\ \text{Ax} = \mathbf{b}, \quad & \mathbf{x} \geq \mathbf{0} \end{aligned}$$

a kanonikus alakú *lineáris programozási feladat*.

A hagyományos lineáris programozási feladatra felírt criss-cross módszerre a végesség első alkalommal [8]-ben került bemutatásra az **s**-monoton indexválasztási szabályok alkalmazása esetén.

1. Eredmény. *Legyen adott egy kanonikus alakú lineáris programozási feladat. Ekkor a criss-cross algoritmus véges az s-monoton indexválasztási szabályok alkalmazása esetén.*

A lineáris komplementaritási feladat a lineáris programozási feladat egy kiterjesztése, melyben változó párok szorzatáról írjuk elő a komplementaritást (szorzatuk nulla). A kanonikus alakú lineáris komplementaritási feladat a következő alakban írható fel, ahol $\mathbf{u}, \mathbf{v} \in \mathbb{R}^n$:

$$\begin{aligned} -M\mathbf{u} + \mathbf{v} &= \mathbf{q}, \\ \mathbf{u} \mathbf{v} &= \mathbf{0}, \mathbf{u}, \mathbf{v} \geq \mathbf{0} \end{aligned}$$

továbbá $M \in \mathbb{R}^{n \times n}$, $\mathbf{q} \in \mathbb{R}^n$ és $\mathbf{u} \mathbf{v} = (u_1 v_1, \dots, u_n v_n) \in \mathbb{R}^n$.

A lineáris komplementaritási feladatra megfogalmazott criss-cross módszerre, a tézis egy némileg általánosabb bizonyítást ad a végességre az **s**-monoton indexválasztási szabályok alkalmazása esetén, melyben nem teszi fel a ciklizáló ellenpélda végességét. Habár ez az általánosítás nem jelentős (és elméletben azonos az eredetivel [3]), mégis közvetlenebbé teszi a kapcsolatot az algoritmus implementációjával, mely közvetlen módon tartalmaz megfontolásokat magából a bizonyításokból. Az EP-tételek szellemében általánosított LCP criss-cross algoritmusra a dolgozat egy Matlab implementáción alapulva mutat be numerikus eredményeket piaci egyensúlyi és bimátrix játékokból adódó LCP feladatokon; az ilyen típusú feladatok különösen érdekesek ezen algoritmus esetén, hiszen mátrixuk nem elégséges. Az eredmények a [3]-ben lettek publikálva.

2. Eredmény. *Az általánosított criss-cross módszer alkalmas nem triviális LCP feladatok megoldására, a feladat mátrixának nem elégséges volta esetén is.*

A kvadratikus primál szimplex módszer végessége

Egy lineáris feltételes konvex kvadratikus feladat esetén, a lineáris célfüggvényt $\mathbf{c} \in \mathbb{R}^n$, míg a kvadratikus részt $Q \in \mathbb{R}^{n \times n}$ jelöli.

$$\begin{aligned} \min \quad & \frac{1}{2} \mathbf{x}^T Q \mathbf{x} + \mathbf{c}^T \mathbf{x} \\ \text{Ax} \leq \mathbf{b}, \mathbf{x} \geq \mathbf{0} \end{aligned}$$

ahol $Q \in \mathbb{R}^{n \times n}$ és $A \in \mathbb{R}^{m \times n}$ mátrixok míg $\mathbf{c} \in \mathbb{R}^n$ és $\mathbf{b} \in \mathbb{R}^m$ megfelelő vektorok.

A lineáris feltételes konvex kvadratikus feladatra felírt primál szimplex módszer a Karush-Kuhn-Tucker feltételekből nyert lineáris komplementaritási feladaton dolgozik:

$$\begin{pmatrix} \mathbf{s} \\ \mathbf{z} \end{pmatrix} - \begin{pmatrix} -A & 0 \\ Q & A^T \end{pmatrix} \begin{pmatrix} \mathbf{x} \\ \mathbf{y} \end{pmatrix} = \begin{pmatrix} \mathbf{b} \\ \mathbf{c} \end{pmatrix}$$

A lineáris feltételes konvex kvadratikus programozásra felírt primál szimplex algoritmusra a tézis a végességet az \mathbf{s} -monoton indexválasztási szabályok mellett bizonyítja. A bizonyítás ennél általánosabb eredményt is bemutat: egy visszavezetésen alapuló gondolatmenet segítségével megmutatja, hogy tetszőleges indexválasztási szabály esetén, mely kizárólag a transzformált jobboldal és a redukált költségek vektorának előjelszerkezetére épül, és melynek alkalmazása esetén a lineáris programozásbeli primál szimplex módszer véges, azokra szükségszerűen a lineáris feltételes konvex kvadratikus programozási primál szimplex algoritmus is véges. A szerző legjobb tudomása szerint, ez az első ilyen jellegű eredmény a kvadratikus szimplex módszerre nézve, és korábban annak végességének bizonyítása kizárólag perturbációs megfontolások alapján lett publikálva.

Az algoritmus az úgynevezett majdnem komplementáris bázisok sorozatán dolgozik.

4. Definíció. *Legyen adott egy LCP feladat. A lineáris komplementaritási feladat egy bázisát majdnem komplementárisnak nevezzük, amennyiben komplementáris 2 indexpár kivételével. Vagyis léteznek olyan $\{i, j\} \in \{1 \dots 2n\}$ indexek, hogy $u_k v_k = 0$ teljesül minden $k \in \{1 \dots 2n\} - \{i, j\}$ indexre nézve.*

Két komplementáris bázis között, a kvadratikus primál szimplex módszer tetszőleges számú majdnem komplementáris bázist generálhat.

5. Definíció. *Legyen adott a QP feladat. Az kvadratikus primál szimplex módszer által generált, két egymást követő komplementáris bázis közötti bázistáblák illetve pivotok sorozatát huroknak nevezzük.*

A végesség bizonyítása egy ciklizálási példa pivotmátrixának elemzésén alapul.

3. Eredmény. Tekintsünk egy lineáris feltételes konvex kvadratikus programozási feladatot a hozzá tartozó lineáris komplementaritási rendszerrel. Egy ciklizáló példa esetén, a kvadratikus primál szimplex módszer kizárólag végessok 1 hosszú hurkot végezhet.

A bizonyítás azon alapul, hogy egy 1 hosszú hurok esetén a báziscsere megfelel a keresési irány kvadratikus célfüggvény szerinti minimumának és így nem degenerált. A bázisban levő primál illetve duál változók számának ellemzéséből adódik a következő eredmény.

4. Eredmény. Tekintsünk egy lineáris feltételes konvex kvadratikus programozási feladatot a hozzá tartozó lineáris komplementaritási rendszerrel. Ekkor tetszőleges ciklizálási példa esetén, a kvadratikus primál szimplex módszer legfeljebb végessok nem 2 hosszúságú hurkot végezhet.

A legtöbb ciklizálási példa lineáris programozási szimplex típusú módszerek esetén szükségképpen primál degenerált. Hasonló eredmény igazolható a kvadratikus változat esetén is, amennyiben a pivot táblát a primál részre szorítjuk meg.

5. Eredmény. Tekintsük a konvex kvadratikus programozási feladatot. Egy ciklizáló példa esetén, véges számú pivot művelet után, a bázismegoldások primál része nem változik többet.

A lineáris programozási esetre való visszavezetés számos, a bilineáris mátrixok bázis-transzformációjára ismert eredményt felhasznál.

6. Eredmény. Tekintsük a konvex kvadratikus programozási feladatot. Egy ciklizáló példa esetén, véges számú pivot művelet után, a kvadratikus primál szimplex módszer már csak 2 hosszúságú hurkokat végez, melyekre a következő tulajdonságok igazolhatóak:

- a hurok első báziscseréje degenerált, melynek során egy primál változó belép a bázisba, illetve egy másik primál változó kilép a bázisból.
- a hurok második, végső báziscseréje során, az első báziscserében belépett primál változó duál párja kilép a bázisból, míg az első báziscserében a bázisból kilépett primál változó duál párja belép a bázisba.
- a hurok második báziscseréje során, a belépő duál változó transzformált oszlopában a bázisban levő primál változóknak megfelelő sorokban minden együttható nulla.
- egy komplementáris bázis esetén egy új hurok megkezdésekor, a bejövő primál változó transzformált oszlopában a bázisban levő duál változók sorában az együtthatók értéke nulla.

	\mathcal{N}		\mathcal{M}	
\mathcal{P}	G		$\begin{array}{c} 0 \\ \vdots \\ 0 \end{array}$	d
\mathcal{D}	$\begin{array}{c} 0 \\ \vdots \\ 0 \end{array}$	$-G^T$		f

1. ábra. Egy ciklizálási példa esetén, a kvadratikus primál szimplex módszer bázistáblájának a struktúrája.

A visszavezetés alapja egy tetszőleges ciklizálási példán megmutatni, hogy a bázistábla tartalmaz egy részt, mely megfeleltethető egy kisebb lineáris programozási feladat bázistáblájának melyen az algoritmus ugyanolyan elvek alapján választja a pivot pozíciót mint a nagyobb LCP feladat mátrixán. A megfelelő LCP pivot tábla struktúrája az 1. ábrán látható.

7. Eredmény. *A lineáris komplementaritási feladaton dolgozó lineáris feltételes konvex kvadratikusan programozási feladat véges az s-monoton indexválasztási szabályok alkalmazása esetén.*

Az eredmények [5]-ben lettek publikálva (illetve angol nyelven [7]).

Egy tesztelési keretrendszer

Egy széleskörű, publikusan elérhető, közepesen nehéz feladatokból álló teszthalmaz összeállítása érdekében, a tézis a következő szabadon rendelkezésre álló teszthalmazokból válogatott feladatokat: NETLIB, Miblib 3, Miplib 2010 illetve egy kisebb ipari feladathalmaz melyek munkaerő egyensúlyi és hozzárendelési problémákat fogalmazznak meg.

A pivot módszerek és a rugalmas indexválasztási szabályok összehasonlításának módja nem nyilvánvaló kérdés, hiszen az implementálási részletek drasztikus mértékben befolyásolják az eredményeket. A tézis ezért egy tesztelési irányelv rendszert fogalmaz meg [6] melyek célja az algoritmus specifikus implementálási részletek hatásának minimalizálása az összehasonlítások érdekében végzett számítások során. A főbb megfontolások a következők:

- a feladatok kanonikus alakját használjuk,
- a tesztelést nyilvános teszthalmazokon végezzük, a feladatok a kanonikus alakra hozatala után,
- kizárólag olyan algoritmikus megfontolásokat implementálunk, melyek közel azonos módon alkalmazhatóak az összes vizsgált algoritmus esetén,
- a numerikus algebrai rutinokat egy elterjedt cél-szoftver függvényeivel valósítjuk meg,
- az implementációt a reprodukálhatóság érdekében szabadon elérhetővé tesszük.

A tézis célja egy olyan összehasonlítási keretrendszer alkalmazása, mely az algoritmusok fő tulajdonságaira illetve az indexválasztási szabályok alkalmazhatóságára koncentrálna, függetlenül az implementálás egyéb szempontok szerinti kidolgozottságától. Az egységes tesztkörnyezet kialakítása érdekében az algoritmusok alap változata került implementálásra, azon szempontok figyelembe vétele nélkül melyek általában jelentősen növelnék a módszerek hatékonyságát illetve stabilitását azonban azonos hatékonyságú módon való megvalósítása a különböző módszerekre nehezen megvalósítható.

Rugalmas indexválasztási szabályok numerikus vizsgálata

Két külön implementáció került megvalósításra a numerikus tanulmányok készítése során: egy Matlab megvalósítás a kisebb feladatok vizsgálatára, illetve egy C implementáció az Xpress lineáris algebrai függvényeire építve a tesztalmazók nagyobb feladatainak megoldhatóságának érdekében: a rugalmas indexválasztási szabályok potenciális előnyeit mindkét kísérlet alátámasztja [6].

A Matlab implementáció a criss-cross módszer elemzésére bizonyult hatékonynak, mind a lineáris programozási, mind a lineáris komplementaritási feladatok esetén, ahol az elsődleges szűk keresztmetszet nem a lineáris algebra illetve LCP esetén a vizsgált feladatok mátrixa esetén az elégségesség hiánya volt, hanem az algoritmus végességének kombinatorikus jellege. Ennek eredményeképpen a criss-cross módszer által megoldhatónak bizonyult feladatok mérete meglehetősen kicsi maradt a lineáris programozásban elfogadottan közepes méretűnek gondolt feladatokhoz képest. Az ide vonatkozó eredmények megtalálhatóak [4]-ben.

A criss-cross módszer megvalósításával szemben, a primál szimplex és a monotonic build-up szimplex módszerek esetén, hamar világossá vált hogy a szűk keresztmetszet az algoritmusok numerikus érzékenysége. Itt érdemes megjegyezni, hogy a numerikus problémák hasonló módon fellépnek a cross-cross módszer esetén is, ahol azonban azok jelenléte kevésbé nyilvánvaló hatásokat okoz és közvetlen módon a végességet befolyásolja, míg a szimplex változatok esetén azokat könnyebb detektálni, hiszen a hatás gyakran nyilvánvaló, mint például apró nemnegativitások jelentkezése a transformált jobboldalon annak ellenére hogy a hánycostest figyelembe vételre került, vagy például a célfüggvény értékének a redukált költség által jelzett iránnyal ellentétes irányba való elmozdulása.

8. Eredmény. *A numerikus vizsgálatok igazolják a rugalmas indexválasztási szabályok hatékonyságát és alkalmazhatóságát ciklizálási problémák kezelésére.*

A tézis két, érdekesebb összefoglaló táblázatán látható hogy az MBU szimplex algoritmus relatív egyszerű implementációja kevesebb feladatot oldott meg, mint a hagyományos primál szimplex módszer. Ami nem meglepő, hiszen az MBU egy lényegesen összetettebb eljárás, így nagyobb eséllyel kényszerül numerikus problémákkal szembenézni. Figyelemre méltó, hogy az egyes variánsok összességében mennyire különböző feladatokat voltak képesek megoldani.

Mindez jól mutatja hogy mennyire eltérő pivot sorozatokat végeznek. Fontos észrevétel, hogy habár a hagyományos szimplex módszer láthatóan stabilabb, ennek ellenére nem dominál az MBU változathoz viszonyítva.

	Most negative	Minimál	LIFO	MOSV	Hybrid LIFO	Hybrid MOSV
Szimplex	209	150	168	150	161	175
MBU	150	145	149	149	146	146

1. táblázat. Az algoritmus és indexválasztási szabály kombinációk által sikeresen megoldott feladatok száma a teljes teszthalmazból.

	Most negative	Minimál	LIFO	MOSV	Hybrid LIFO	Hybrid MOSV
Szimplex	84	36	32	40	48	69
MBU	31	25	31	28	30	31

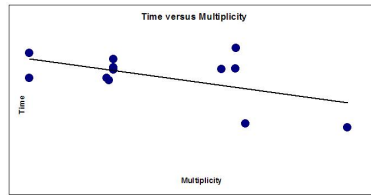
2. táblázat. Azon esetek száma, ahol az algoritmus és indexválasztási szabály kombináció a leggyorsabbnak bizonyul. A torzítás csökkentésének érdekében az idő értékek másodpercre kerekítettek, így számos feladat több helyen is elszámolásra került.

A legjobb eredményt a hagyományos szimplex módszer érte el a legnegatívabb szabály esetén. Ez az érték összehasonlítási referenciaként van bemutatva hiszen elméleti szempontból nem véges eljárás. A megoldott feladatok számát mérő teszt elsősorban az eljárások hatékonyságát méri, így nem meglepő hogy az a módszer bizonyul a leghatékonyabbnak mely a legkisebb számú báziscserét végzi. Habár ez eredmény nem meglepő, a számok relatív nagyságának mértéke nagyobb mint az eredetileg várható volt. Érdekes, hogy a legnegatívabb szabály kisebb mértékben dominál az MBU esetén; ez az extra duál hányadoseszt hatása, mely gyakran felülírja az eredetileg választott bejövő oszlopot. A rugalmas indexválasztási szabályok flexibilitása és hatékonysága közötti kapcsolat mérésének érdekében, a tézis bevezeti az indexválasztási szabályok rugalmassági fokának a fogalmát.

6. Definíció. *Egy szimplex típusú módszer esetén, adott feladatra és indexválasztási szabályra nézve, az indexválasztási szabály rugalmassági fokán az összes iteráció során végzett báziscserék esetén azon indexek számának összegét értjük, melyek közül az indexválasztási szabály szabadon hagyta a tényleges pivot pozíció meghatározását.*

A vizsgált feladatokon, a rugalmassági fok és a teljes megoldási idő összefüggését a következő grafikon szemlélteti.

A szerző tudomása szerint az MBU és a hagyományos szimplex numerikus összehasonlítása az első alkalommal került publikálásra, így az eredmény önmagában is érdekes.



2. ábra. A rugalmassági fog és a megoldási idő összefüggése.

9. Eredmény. Az MBU szimplex módszer hatékonysága jól mutatja annak versenyképességét a hagyományos szimplex algoritmussal szemben, gyakorlati alternatívát kínálva.

A tézis alapjául szolgáló publikációk

A LCP criss-cross numerikus eredmények a [3]-ben lettek bemutatva. A kvadratikus primál szimplex algoritmus végessége s -monoton szabályok esetén (beleértve az általános visszavezetést) az [5, 7]-ban került publikálásra. A LP criss-cross módszer végességét s -monoton szabályok esetén a [8]-es tartalmazza. A Matlab implementáción alapuló eredményeket a primál szimplex és az MBU szimplex hatékonyságáról rugalmas szabályok alkalmazása esetén a [2] dolgozat tartalmazza. A tesztelési keretrendszer, illetve a kiterjedt numerikus vizsgálat a primál szimplex és az MBU szimplex módszereknek rugalmas indexválasztási szabályok esetén, egy kereskedelmi megoldó lineáris algebráját használva a [6]-ben lettek közzéadva.

Hivatkozások

- [1] Zs. Csizmadia. *New pivot based methods in linear optimization, and an application in petroleum industry*. PhD thesis, Eötvös Loránd University of Sciences, 2007.
- [2] Zs. Csizmadia, T. Illés, and A. Nagy. The s -monotone index selection rules for pivot algorithms of linear programming. *European Journal of Operation Research*, 221(3):491–500, 2012.
- [3] Zs. Csizmadia, T. Illés, and A. Nagy. The s -monotone index selection rule for criss-cross algorithms of linear complementarity problems. *Acta Universitatis Sapientiae Informatica*, 5(1):103–139, 2013.
- [4] T. Illés and A. Nagy. Lineáris programozási szoftverek tesztelése. In *Informatika a felsőoktatásban 2008: IF2008 Konferencia kötete*, 2007.
- [5] T. Illés and A. Nagy. A kvadratikus szimplex algoritmus végessége indexválasztási szabályok alkalmazása esetén. *Alkalmazott Matematikai Lapok*, 30:1–21, 2013.
- [6] T. Illés and A. Nagy. Computational aspects of simplex and mbu-simplex algorithms using different anti-cycling pivot rules. *Optimization*, 63(1):49–66, 2014.
- [7] T. Illés and A. Nagy. Finiteness of the quadratic primal simplex method when s -monotone index selection rules are applied. 2014. Operations Research Report, 2014-01.
- [8] A. Nagy. Finiteness of the criss-cross method for the linear programming problem when s -monotone index selection rules are applied. 2014. Operations Research Report, 2014-02.